

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний технічний університет
імені Івана Пулюя

Кафедра автоматизації
технологічних
процесів та виробництв

Методичні вказівки
для виконання лабораторної роботи №1 “Обробка
зовнішніх переривань по входу RB0 / INT на
програмному симуляторі PIC Simulator IDE”
з курсу “Проектування мікропроцесорних систем
керування технологічними процесами”

Тернопіль 2017

Методичні вказівки для виконання лабораторної роботи №1 «Обробка зовнішніх переривань по входу RB0/INT на програмному симуляторі PIC Simulator IDE» з курсу «Проектування мікропроцесорних систем керування технологічними процесами».

Методичні вказівки розглянуті і схвалені кафедрою «Автоматизація технологічних процесів та виробництв», протокол №4 від 21.11.2016 р.

Відповідальні за випуск

доцент, к.т.н. Медвідь В.Р.,
асистент Пісьціо В.П.

Лабораторна робота №1

Обробка зовнішніх переривань по входу RB0 / INT на програмному симуляторі PIC Simulator IDE

1. Робота з програмним симулятором PIC Simulator IDE

Запустивши на виконання PIC Simulator IDE, побачимо основне вікно цієї програми (рис. 1).

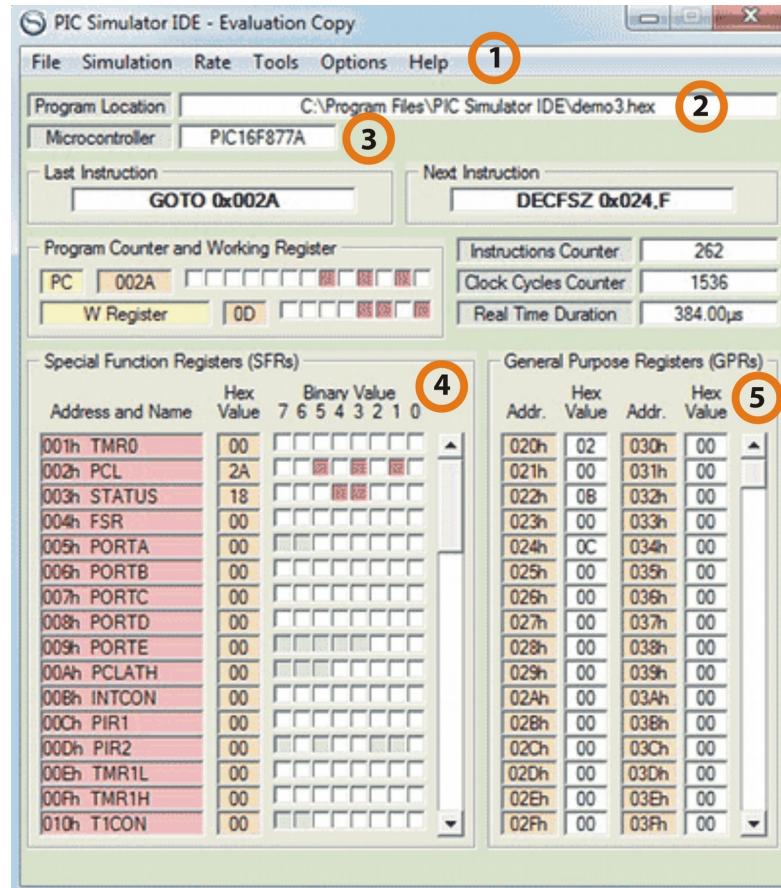


Рис. 1. Основне вікно програми PIC Simulator IDE

У верхній частині знаходяться різні меню, через які можна отримати доступ до різних основних і додаткових модулів програми (на рис. 1 позначено як «1»).

Далі, в рядку Program Location вказано шлях до обраної програми і її ім'я (на рис. 1 - «2»).

Нижче, в рядку Microcontrollers, відображається тип обраного мікроконтролера (на рис. 1 - «3»).

У нижній частині вікна є дві панелі (позначені як «4» і «5»). У них відображаються стан програми, вміст спеціальних і керуючих регістрів обраного МК.

Послідовність роботи з програмним симулятором наступний:

- запуск програми PIC Simulator IDE;
- вибір типу мікроконтролера, для якого написана програма;
- вибір частоти кварцового генератора (впливає тільки на відображувані програмою дані про час виконання програми або команди, але не на швидкість роботи програми, що налагоджуються в PIC Simulator IDE);
- завантаження програми у вигляді HEX-файлу або запуск вбудованого компілятора мови assembler і написання в ньому потрібної програми;
- вибір потрібних модулів віртуальних пристроїв;
- вибір швидкості і режиму роботи програми симулятора;

- запуск процесу симуляції роботи програми на обраному МК.

Якщо потрібно скористатися для роботи з симулятором власною програмою або внести зміни у вже розроблену, необхідно створити або завантажити для цього файл асемблера, з якого після компіляції буде створений необхідний для роботи з симулятором hex-файл.

Для цього:

1. Натиснути Options | Assembler. Відкриється вікно компілятора Assembler – UNTITLED (рис. 2);

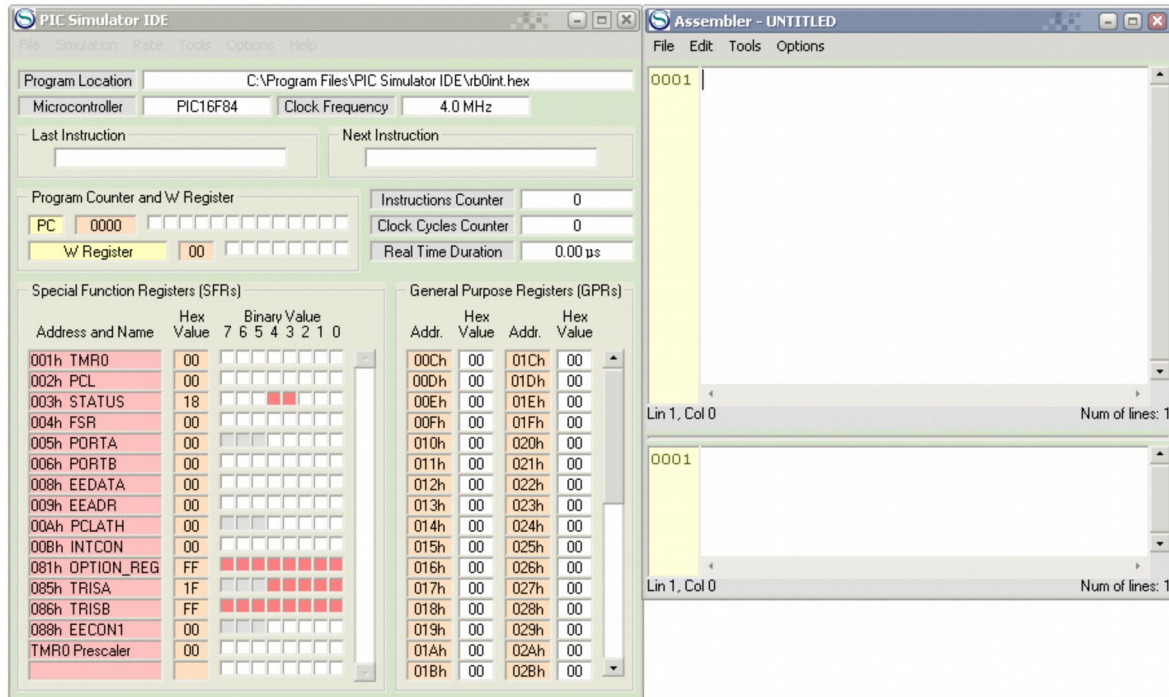


Рис. 2 Вікно симулятора з відкритим вікном Assembler

2. У вікні Assembler натисніть опцію File. Розкриється закладка (рис. 3), з якої для створення нового файлу потрібно натиснути New, а для завантаження вже створеного – OPEN.

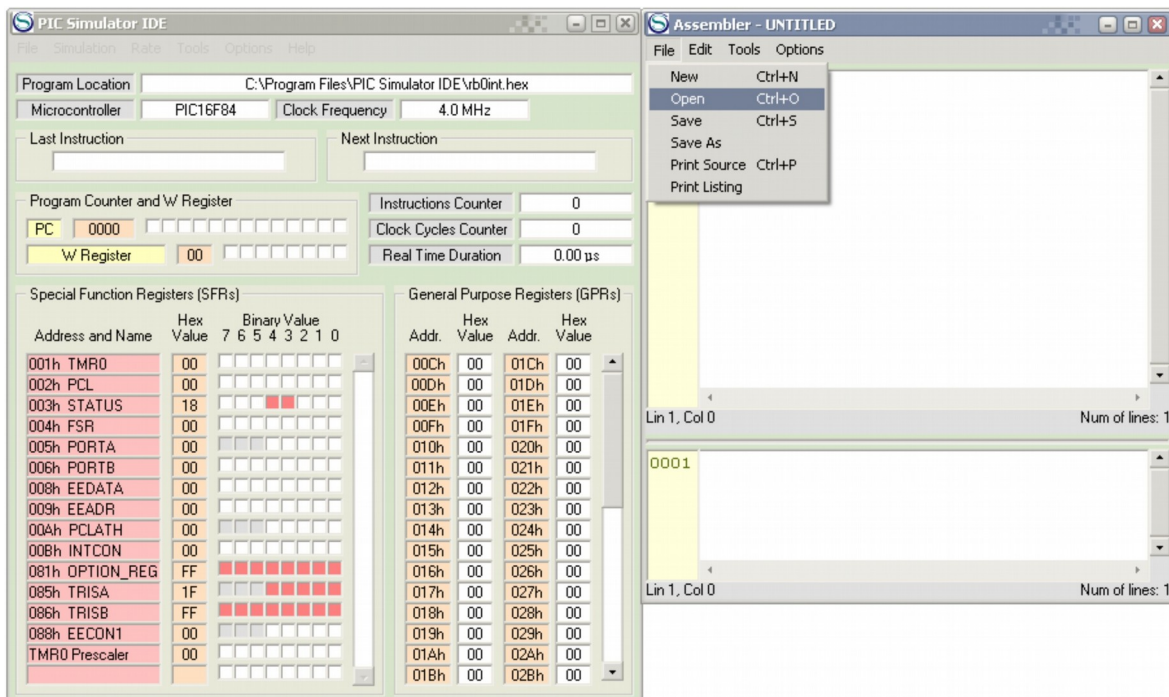


Рис. 3

3. Після вибору і завантаження файлу (наприклад. rb0int.asm), його текст з'явиться в верхній половині вікна Assembler (рис. 4).

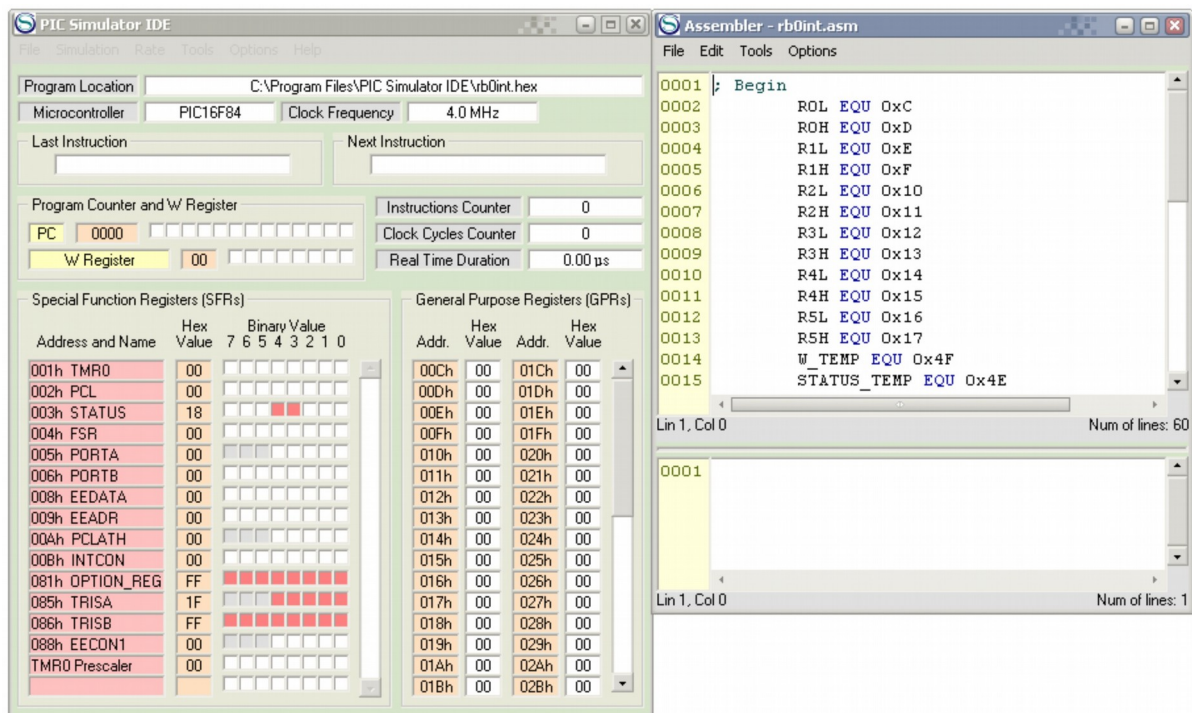


Рис. 4 Завантаження файлу rb0int.asm

4. Для компіляції створеного або завантаженого і потім зміненого файлу, натисніть Tools і у вікні, що розкриється – Assemble. В нижній половині вікна Assembler з'явиться відкомпільований файл і одночасно, при відсутності помилок, буде створений одноіменний hex-файл.

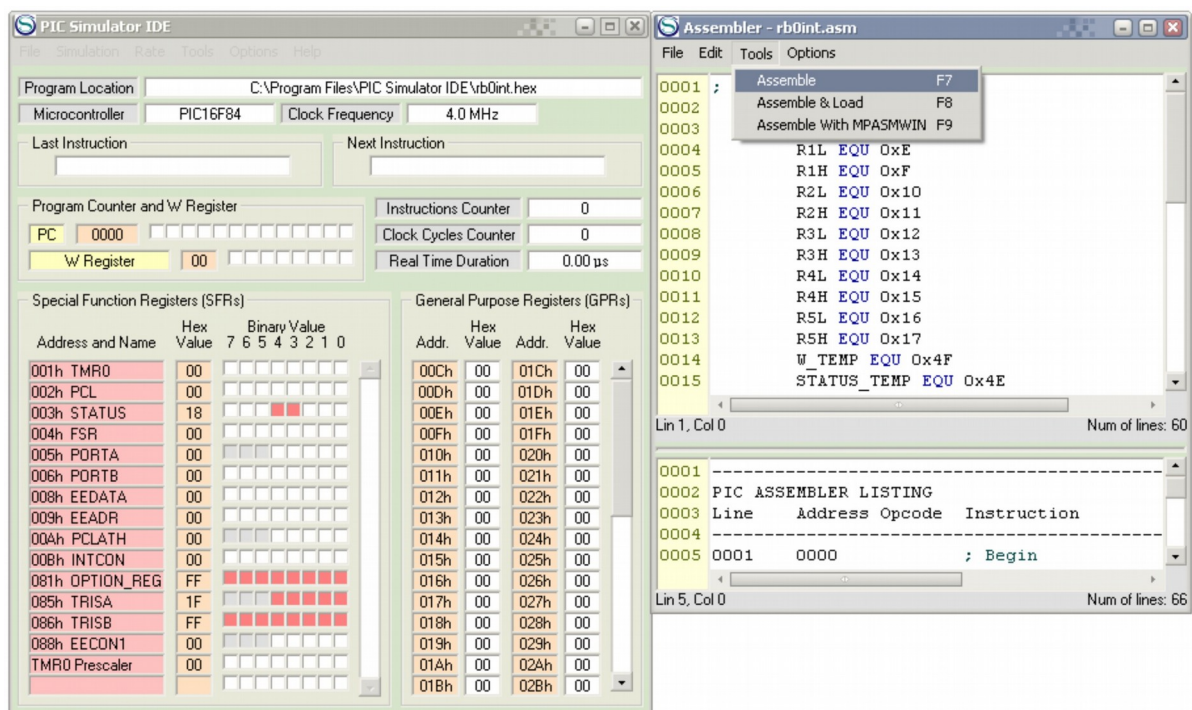


Рис. 5

2. Завдання на лабораторну роботу: обробка зовнішніх переривань мікроконтролера по входу RB0 / INT.

1. Вивчити програмну модель PIC Simulator IDE.
2. Вивчити команди обміну даними через порти PIC – контролера.
3. Написати і дослідити роботу програми з Прикладу 1 та дослідити вміст регістрів контролера, які використовуються при виконанні цієї програми.
4. Користуючись вікном “Program Memory Editor” записати перші 7 команд виконуваної програми на асемблері та в шістнадцяткових кодах.
5. Записати для вибраних команд коментар щодо їх призначення (див. Приклад 2).

Приклад 1

Ця програма змінює значення порту PORTA при зміні стану входу RB0 / INT по фронту імпульсу.

Текст програми з файлу rb0int.asm має наступний вигляд:

```
; Begin
R0L EQU 0xC
R0H EQU 0xD
R1L EQU 0xE
R1H EQU 0xF
R2L EQU 0x10
R2H EQU 0x11
R3L EQU 0x12
R3H EQU 0x13
R4L EQU 0x14
R4H EQU 0x15
R5L EQU 0x16
R5H EQU 0x17
W_TEMP EQU 0x4F
STATUS_TEMP EQU 0x4E
ORG 0x0000
BCF PCLATH,3
BCF PCLATH,4
GOTO L0001
ORG 0x0004
MOVWF W_TEMP
SWAPF STATUS,W
CLRF STATUS
MOVWF STATUS_TEMP
CALL L0002
SWAPF STATUS_TEMP,W
MOVWF STATUS
SWAPF W_TEMP,F
SWAPF W_TEMP,W
RETFIE
L0001:
; 1: TRISA = 0x00 'set all PORTA pins as outputs
BSF STATUS,RP0
CLRF 0x05
BCF STATUS,RP0
; 2: PORTA = 0xff 'make all PORTA pins high
MOVLW 0xFF
```

```

MOVWF 0x05
; 3: INTCON.INTE = 1 'enable RB0/INT interrupts
BSF 0x0B,4
; 4: INTCON.GIE = 1 'enable all un-masked interrupts
BSF 0x0B,7
; 5: End
L0003: GOTO L0003
; 6:
; 7: On Interrupt 'interrupt routine
L0002:
; 8: PORTA = PORTA - 1 'decrement the value on PORTA
MOVLW 0x01
SUBWF 0x05,W
MOVWF 0x05
; 9: INTCON.INTF = 0 'enable new RB0/INT interrupts
BCF 0x0B,1
; 10: Resume
RETURN
; End of program
L0004: GOTO L0004
; End of listing
END

```

3. Послідовність роботи з симулятором при виконанні програми

Запустімо цю програму в PIC imulator IDE в такій послідовності:

1. Запустити PIC Simulator IDE;
 2. Натиснути Options | Select Microcontroller;
 3. Вибрати PIC16F84 і натиснути кнопку Select;
 4. Натиснути File | Load Program;
 5. Вибрати файл rb0int.hex і натиснути кнопку Open (програмне забезпечення завантажено);
 6. Натиснути Tools | Microcontroller View (відкриється вікно Microcontroller View із схематичним зображенням мікроконтролера та його виводів, а також станом виводів ON/OFF);
 7. Вибрати Rate | Step By Step (покроковий режим виконання програми);
 8. Натиснути Tools | Program Memory Editor (відкриється вікно з виконуваною програмою на асемблері та в кодах - бінарному та шістнадцятковому);
 8. Натиснути Simulation | Start (почнеться виконання завантаженої програми). Причому, програма буде виконуватися по одній команді. Для виконання кожної команди необхідно натиснути на закладку STEP, яка з'явиться вгорі основного вікна після натискання "Start";
 9. Для виконання програми в автоматичному режимі потрібно вибрати Rate | Extremely Fast simulation rate.
 10. Щоб зупинити виконання програми, потрібно натиснути Simulation | Stop.
- Вигляд екрану з виконуваною програмою показано на рис. 2 (натискання кнопки «Т», пов'язаної з ніжкою RB0 / INT, переключить логічний стан цього виводу).
- Вигляд екрану з виконуваною програмою «Обробка зовнішніх переривань на прикладі RB0/INT» показано на рис. 6.

Вміст регістрів контролера, які використовуються при виконанні програми, знаходимо з області регістрів Address and Name , яка розташована в лівій нижній частині основного вікна симулятора (виділені рожевим кольором) (рис. 2). Всі регістри возьмирозрядні.

В процесі виконання програми по зміні кольору комірок видно, вміст яких регістрів змінюється. Забарвлення комірки відповідного розряду регістра помаранчевим кольором означає наявність “1”, білим - “0”. Вміст регістрів записуємо в шістнадцятковому коді за Прикладом 3.

Приклад 3

Регістр	Вміст регістру
PORTA	1F
і т.д.	

4. Контрольні запитання

1. Система команд мікроконтролера.
2. Структура портів мікроконтролера.
3. Програмування портів мікроконтролера.
4. Види переривань мікроконтролера.
5. Призначення та позначення основних елементів програмної моделі мікроконтролера.

5. Література

1. Данилин А. Программа-симулятор PIC Simulator IDE / Данилин А. // Современная электроника. 2006.- №4. -С. 68-76.
2. Тавернье К. PIC-микроконтроллеры. Практика применения. М.: ДМК, 2002.
3. Предко М. Создайте работа своими руками на PIC- контроллере./ Майкл Предко; Пер. с английского Земского Ю.В. – М.: ДМК Пресс, 2006. – 408 с.: ил. – (В помощь радиолюбителю).
4. Кениг А. и М. Полное руководство по PIC-микроконтроллерам.: Пер. с нем.-К.: “МК-Пресс”, 2007.-256 с., ил.